THIS PAGE BLANK (USPTO)

XP-000779509

IEICE TRANS. COMMUN., VOL. E81-B, NO. 5 MAY 1998

PAPER

An Approach for Cluster-Based Multicast Routing in Large-Scale Networks

Yibo ZHANG†, Weiping ZHAO††, Shunji ABE††, *and* Shoichiro ASANO††, *Members*

SUMMARY This paper addresses the optimum routing problem of multipoint connection in large-scale networks. A number of algorithms for routing of multipoint connection have been studied so far, most of them, however, assume the availability of complete network information. Herein, we study the problem under the condition that only partial information is available routing nodes and that routing decision is carried out in a distributed cooperative manner. We consider the network be-partitioned into clusters and propose a cluster-based routing approach for multipoint connection. Some basic principles for network clustering are discussed first. Next, the original multipoint routing problem is defined and is divided into two types of subproblems. The global optimum multicast tree then can be obtained asymptotically by solving the subproblems one after another iteratively. We propose an algorithm and evaluate it with computer simulations. By measuring the running time of the algorithm and the optimality of resultant multicast tree, we show analysis on the convergent property with varying network cluster sizes, multicast group sizes and network sizes. The presented approach has two main characteristics, 1) it can yield asymptotical optimum solutions for the routing of multipoint connection, and 2) the routing decisions can be made in the environment where only partial information is available to routing nodes.

key words: multicast routing, multipoint connection, large-scale network, clustering, global optimization, aggregate/disaggregate flow

1. Introduction

The advent of advanced switching technologies makes it possible to realize efficient multipoint connection, which has applications in many emerging communication-based systems, e.g., video on demand service, teleconferencing and distant education. Observing that the multipoint connection may occupy much network resources with long duration, the transmission routes used for the connection have to be selected such that they consume minimum network resources. Tree-shaped routes (i.e., *multicast trees*) that connect the source and destinations usually serve as the candidates for such communications.

The optimization objective for the routing of multipoint connection is to minimize the multicast tree cost, which is the sum of link costs of the multicast tree. The minimum-cost tree is termed Steiner tree, finding such a tree is known to be NP-complete. The Steiner tree problem has first been studied in the area of graph theory. A variety of algorithms both for finding exactly and approximately optimum solutions have been proposed so far [1], of which the heuristic algorithms are beneficial to real-time applications because they find approximate solutions within polynomial time. In the study of realization of multipoint connection in communication networks, some concrete issues have been considered in the literature. For instance, [3] has studied the multipoint connection problem where destinations change during the life of the connection. A similar solution using quasi-static method based on statistical traffic pattern has been presented by [6]. Delay-constrained multicast routing has been examined by [5], [7]. [4] has investigated the routing problem for multipoint multi-stream connection. Generating multicast trees in networks with directed, different characteristic links was presented in [8]. Instead of enumerating all of the previous work, our goal here is trying to find the problem which has been ignored ever before. Note that most of the existing algorithms are suitable for centralized implementation due to their implicit assumption that complete information of the network is given, they may bring with drawbacks of not scaling well or might even be not applicable for large-scale networks. Thus, certain techniques which can deal with global optimization objective efficiently under the condition of partial information are of necessity. In this aspect, some investigation can be found in [2], where, however, only shortest-path based multicast routing has been considered.

In this paper, we focus on the routing approach which can achieve global optimum solutions in large-scale networks. A general routing problem requires us to make optimum routing decisions quickly in the circumstance of changing traffic and resource information about the network. We assume that the network information is managed locally by individual clusters, and set the objective as to find global optimum multicast tree for multipoint connection. The idea of clustering has been adopted in the routing of conventional point-to-point connection, and is known effective for fast routing [10], [11]. New task occurs for the routing of multipoint connection as their objective function definitions differ from those of point-to-point connection. We call the problem of finding global optimum

Manuscript received August 4, 1997.
Manuscript revised December 4, 1997.

†The author is with the Faculty of Engineering, University of Tokyo, Tokyo, 113-0033 Japan.
††The authors are with the National Center for Science Information Systems, Tokyo, 112-0012 Japan.

multicast trees in such an environment as cluster-based multicast routing (CBMR) problem. Comparing with its general counterparts, the CBMR is more complicated due to the lack of complete network information. To solve the CBMR problem, we first divide the problem into two types of subproblems, which are generated by separating intra-cluster and inter-cluster flows. Aggregate and disaggregate flows are used to define the subproblems. Each subproblem is formulated as integer linear programming (ILP) problem and then be solved by using linear programming (LP) method. The solution is obtained in a convergent way by solving the subproblems one after another iteratively. Routing decisions are made in a distributed cooperative manner. That is, multiple routing nodes, each of which corresponds to one cluster, may be involved in the routing decisions. Due to the asymptotical feature in finding optimum solutions, trade-offs will be existing between the running time of algorithm and the optimality of the obtained solutions, and can be utilized according to performance requirement. We evaluate the algorithm with computer simulations, and show analysis on the convergent property with varying network cluster sizes, multicast group sizes and network sizes. We also compare our algorithm with one conventional algorithm.

The remainder of this paper is organized as follows. Section 2 provides a brief explanation on principles for network clustering, describes how to generate object graph, and specifies the general form of cluster-based multicast routing problem. Section 3 defines the problem reformulation, presents the corresponding algorithm for multicast routing. Section 4 is dedicated to numerical analysis with computer simulations. Finally, Sect. 5 shows concluding remarks.

2. The Cluster-Based Multicast Routing

2.1 Network Clustering

Given a large-scale network, for general purpose, we assume the network is: (1) not regular topology, and (2) not fully connected.

The network clustering is being considered according to the following two simple requirements: first, making routing decisions fast, and second, managing the network traffic and resource information efficiently. We will pay more attention to the second requirement for the reason that enormous amount of network traffic and resource information need to be handled for large-scale and broadband networks and that multipoint connection will probably consume large amount of network resources. After network clustering, the overall network is partitioned into small subnetworks, namely, clusters. Hereafter, we use *cluster* to represent the management unit for network resources. The resource information of one cluster is available to the routing node(s) of this cluster, but is not available to those of any other clusters. Any two clusters, however, share the resource information of any links connecting these two clusters.

The network are divided into clusters such that clusters are not overlapping, which means no node is shared by any two clusters. This situation can always be retained because a shared node of two clusters can be divided into two pseudo-nodes belonging to the two clusters respectively and being connected by an artificial link with infinite capacity. The cluster size, indicated by the number of nodes in the cluster, is an important parameter which should be determined by three factors. The first is information storage and processing capacity possessed by the routing nodes. The second is the time limit for network information propagation in the interval of two continual routing decisions. The third is the issues stemmed from optimum routing consideration. Obviously, when cluster size is large, the routing nodes need to have high storage and processing capacity, the transmissions of network information will cost highly, and the global optimum solution of routing is likely to be achieved. There are two extreme instances that the cluster size equals to 1 or N (where N is the total number of nodes in the network). It is not straightforward to develop efficient routing approaches for large-scale networks under either of these instances. We have to exclude these instances from consideration.

As this paper focuses on the routing method, we simplify the discussion of clustering method. It should be noted that our routing method does not impose requirement on the clustering procedure. The network clustering can be carried out depending on the value of given cluster size, the connectivity of node and/or the distance between nodes. A basic property that must be guaranteed is that no disjointed node is allowed to appear in any cluster. As a result of clustering, dense connectivity among clusters can be achieved, even for sparse networks.

In the cluster-based model, one or multiple routing nodes of each cluster play the role of both making routing decisions within the cluster, and negotiating with their coteries of other clusters if the route will probably traverse those clusters. Note that each of these nodes has partial information of the network, i.e., it possesses only local information of the cluster it belongs to. Based on the local information, each cluster provides minimum cost path information to adjacent clusters, thus, each cluster can estimate minimum cost paths from it to other clusters.

2.2 Generating Object Graph

In a large-scale network, usually only a portion of clusters contain member(s) of certain multicast group. We create an object graph from the original network graph by eliminating the clusters that do not have multicast member(s) (as shown in Fig. 1). The boundary nodes of different clusters which were connected with the elim-

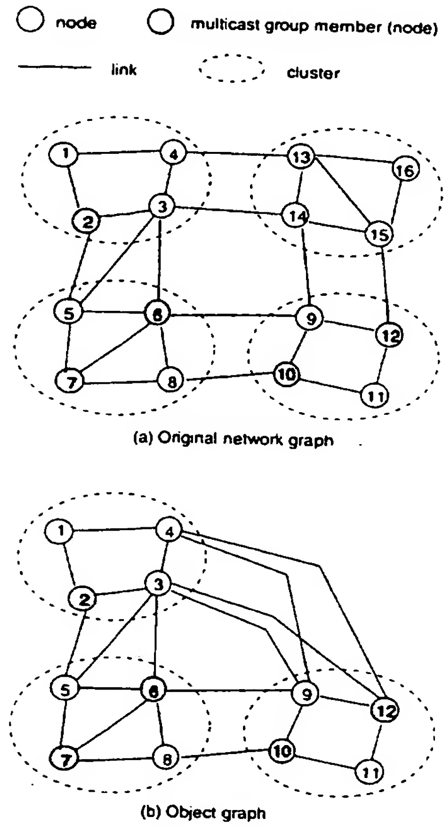

Fig. 1 An original network graph and its object graph.

inated clusters are connected with new *links*. The costs of new links are equal to the costs of shortest paths between corresponding boundary nodes. A shortest path can be calculated by using conventional shortest path algorithms, i.e. Bellman-Ford algorithm [11]. The generated object graph is used as the base for solving the routing problem in later sections.

2.3 Problem Formulation

The object graph is represented by $G = (V, E)$, where V denotes the set of nodes, E denotes the set of links; each link $(i, j) \in E$ connecting nodes i and j is associated with capacity of b_{ij}. The costs imposed for unit flow over the links are determined by a cost function which is predetermined. The value of such a cost c_{ij} $((i, j) \in E)$ is given and used to calculate the cost for a flow over certain path. In the cost structure, only link costs are considered while other costs such as node related costs (e.g., switching costs) are ignored herein. In the object graph, there are g clusters, denoted by $\Theta = \{\rho_1, \rho_2, \cdots, \rho_g\}$. Given a set of nodes, $Z = \{s, Z_d\}$, where s is the source, Z_d is the set of destinations in the multicast group, and given the bandwidth, f, required by the connection, the optimization problem can be formulated as follows.

$$\text{Min} \left\{ \sum_{\substack{i,j\in\rho_r,\rho_r\in\Theta \\ (i,j)\in E}} c_{ij}x_{ij} + \sum_{\substack{i\in\rho_r,\rho_r\in\Theta \\ j\in\rho_q,\rho_q\in\Theta \\ r\neq q,(i,j)\in E}} c_{ij}x_{ij} \right\} \quad (1)$$

subject to

$$x_{ij} \geqq x_{ij}^k + x_{ji}^k \qquad \forall\ (i,j) \in E,\ \forall k \in Z_d \quad (2a)$$

$$\sum_{(i,j)\in E} x_{ij}^k - \sum_{(h,i)\in E} x_{hi}^k = \begin{cases} 1, & \text{if } i = s \\ -1, & \text{if } i = k \\ 0, & \text{otherwise} \end{cases}$$

$$\forall i \in V, \forall k \in Z_d \quad (2b)$$

$$x_{ij} \in \{0, 1\} \qquad \forall\ (i,j) \in E \quad (2c)$$

$$x_{ij}^k \geqq 0 \qquad \forall\ (i,j) \in E, \forall k \in Z_d \quad (2d)$$

$$x_{ij} \leqq \lfloor b_{ij}/f \rfloor \qquad \forall\ (i,j) \in E\ . \quad (2e)$$

The expression of $\lfloor y \rfloor$ in (2e) means the maximum integer less than or equal to y.

In the objective function, the first and second items are the sums of intra-cluster flow costs and inter-cluster flow costs respectively in multicast tree. Constraints (2a) imply that the link used by any path from the source to one destination must be included in the resultant tree. Constraints (2b) describe the transmisson condition for any node according to the position of this node on the path from the source to a destination. Constraints (2c) imply that any link may or may not be contained in the resultant tree, and (2d) describe the similar situation for a link as an element of the path from source to destination. The constraints (2a) to (2d) ensure that each possible route is a tree which connects all the nodes in Z. Constraints (2e) ensure that the possible tree is constituted with those links having enough available capacity. In the above formulation, the problem is described as a 0-1 integer programming problem. It can also be viewed as a special kind of multicommodity flow problem where each flow corresponds to a path from the source to one of the destinations.

3. Solving the Global Optimum Multicast Routing Problem

In this section, we present an approach to solving the problem described in Sect. 2.3. Because the information about a cluster is available to the local routing node(s) of this cluster, it's generally impossible to obtain global optimum multicast routing solutions without negotiating with the routing nodes of other clusters. We employ the decomposition method for the object graph, and apply asymptotic technique to find multicast trees. The problem is decomposed and reformulated in terms of aggregate flows. This is a common idea extensively used for solving flow optimization problems of large networks [12]–[14]. In particular, we extend the idea

generally used for solving minimum-cost flow problems to solve the optimum multicast routing problem in this paper.

3.1 Aggregate Flow Based Problem Decomposition

There are two types of flows, i.e., the intra-cluster flows and inter-cluster flows, existing in the object network. Recall that the routing node for any cluster only possesses the information about the links in the cluster and the links connected with other clusters. A cluster can be viewed as a "node" (***aggregate node***) by other clusters, and the links between a node in the cluster and a aggregate node are then aggregated into one single "link" (***aggregate link***), as illustrated in Fig. 2. In the graph, a dummy node is introduced to ensure problem feasibility and to reduce the complexity in representing potential links between aggregate nodes. For an object graph which contains m clusters, we will have m graphs as depicted in Fig. 2.

Additionally, flows between two clusters can be observed independently by splitting these clusters off, as illustrated in Fig. 3. The links between two clusters are intended to accomodate the possible aggregate flows on the aggregate links in Fig. 2. These links are directional with the direction from the left cluster to the right one. Thus, two such graphs exist for any two connected clusters. The total number of such graphs depends on the connectivity among clusters. Note that only those nodes sharing link(s) with some node(s) in the other cluster are contained in the graph. Two dummy nodes are added, each of which connects bidirectionally with nodes in one of the two clusters.

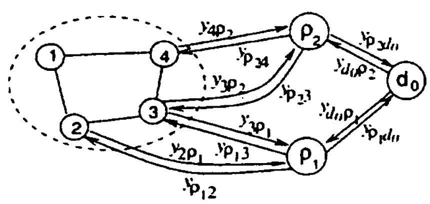

Fig. 2 A graph with one cluster and aggregate nodes.

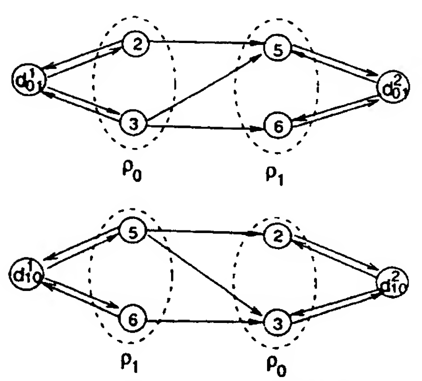

Fig. 3 Two graphs with two connected clusters.

3.2 Problem Reformulation

Two categories of subproblems can be derived from the above decomposition. They correspond to the two types of graphs shown in the previous subsection. The first type of subproblem is to solve the routing problem in one cluster and the associated aggregate nodes while the second type of subproblem is to solve the routing problem between clusters. We use P_1 and P_2 to identify two types of subproblems respectively and give their formulations later.

Prior to proceeding, let us assign the to-be-used notations as follows. (those having been used previously are not repeated)

$y_{i\rho_q}$: Flow on aggregate link (i, ρ_q)

$y_{\rho_q j}$: Flow on aggregate link (ρ_q, j)

$\boldsymbol{y}_{rq}$: $\{y_{i\rho_q}, y_{\rho_r j} | i \in \rho_r, j \in \rho_q\}$

$u_{i\rho_q}$: Cost for unit flow on aggregate link (i, ρ_q)

$u_{\rho_q i}$: Cost for unit flow on aggregate link (ρ_q, i)

$\boldsymbol{u}_r$: $\{(u_{i\rho_q}, u_{\rho_q i}) | i \in \rho_r, q = 1, 2, \cdots, m, q \neq r\}$

d_r: Dummy node for the clusters that are connected with cluster r in subproblem P_1

d^1_{rq}: Dummy node connected with cluster r in subproblem P_2

d^2_{rq}: Dummy node connected with cluster q in subproblem P_2

$v_{i\rho_q}$: Dual variable associated with aggregate link (i, ρ_q)

$v_{\rho_q i}$: Dual variable associated with aggregate link (ρ_q, i)

$v_{d^1_{rq}}$: Dual variable associated with dummy node d^1_{rq}

$v_{d^2_{rq}}$: Dual variable associated with dummy node d^2_{rq}

w_{ij}: Dual variable associated with the flow on link (i, j) between two clusters

$w_{id^1_{rq}}$: Dual variable with link (i, d^1_{rq})

$w_{d^2_{rq} j}$: Dual variable with link (d^2_{rq}, j)

$\boldsymbol{c}$: $\{c_{ij} | i, j \in \rho_r, r = 1, 2, \cdots, m\}$

$\boldsymbol{x}$: $\{x_{ij} | i, j \in \rho_r, r = 1, 2, \cdots, m\}$

$\boldsymbol{v}$: $\{(v_{i\rho_q}, v_{\rho_q i}) | i \in \rho_r; r, q = 1, 2, \cdots, m; r \neq q\}$

$\boldsymbol{y}$: $\{(y_{i\rho_q}, y_{\rho_q i}) | i \in \rho_r; r, q = 1, 2, \cdots, m; r \neq q\}$

$\boldsymbol{w}$: $\{(w_{ij}|i \in \rho_r, j \in \rho_q; r, q = 1, 2, \cdots, m; r \neq q), (w_{id^1_{rq}}, w_{d^2_{rq}j}|i \in \rho_r, j \in \rho_q, r \neq q; r, q = 1, 2, \cdots, m)\}$

$\boldsymbol{b}$: $\{(\lfloor b_{ij}/f \rfloor|i \in \rho_r, j \in \rho_q; r, q = 1, 2, \cdots, m; r \neq q), (1|i \in \rho_r, j \in \rho_q; r, q = 1, 2, \cdots, m); r \neq q\}$

$\rho_{r \to q}$: The set of boundary nodes in cluster ρ_r connected with cluster ρ_q

$\rho_{q \to r}$: The set of boundary nodes in cluster ρ_q connected with cluster ρ_r.

$\boldsymbol{v_q}$: $\{(v_{i\rho_r}, v_{\rho_r i})|i \in \rho_q, r = 1, 2, \cdots, m, r \neq q\}$

$\boldsymbol{w_q}$: $\{(w_{ij}|i \in \rho_q, j \in \rho_r; r = 1, 2, \cdots, m; r \neq q), (w_{id^1_{qr}}, w_{d^2_{qr}j}|i \in \rho_q, j \in \rho_r, r \neq q; r = 1, 2, \cdots, m)\}$

$\boldsymbol{x_r}$: $\{x_{ij}|i, j \in \rho_r\}$

$\boldsymbol{y_r}$: $\{(y_{i\rho_q}, y_{\rho_q i})|i \in \rho_r; q = 1, 2, \cdots, m; q \neq r\}$

$\boldsymbol{v^t_q}, \boldsymbol{w^t_q}, \boldsymbol{x^t_r}, \boldsymbol{y^t_r}$: Variable sets associated with $\boldsymbol{v_q}, \boldsymbol{w_q}, \boldsymbol{x_r}, \boldsymbol{y_r}$ respectively (indicating the results of the t-th iteration of the routing algorithm)

$\boldsymbol{v_q}(t), \boldsymbol{w_q}(t), \boldsymbol{x_r}(t), \boldsymbol{y_r}(t)$: Variable sets associated with $\boldsymbol{v_q}$, $\boldsymbol{w_q}, \boldsymbol{x_r}, \boldsymbol{y_r}$ respectively (indicating the results of the first t iterations of the routing algorithm).

Suppose $\boldsymbol{u_r}$ is given, the objective function of P_1 is to minimize the sums of intra-cluster flow costs of cluster r and the inter-cluster flow costs on aggregate links between the nodes of cluster r and other clusters.

$P_1(r, \boldsymbol{u_r})$:

$$\text{Min} \left\{ \sum_{\substack{i,j \in \rho_r \\ (i,j) \in E}} c_{ij} x_{ij} + \sum_{\substack{i \in \rho_r \\ \rho_q \in \Theta \setminus \{\rho_r\}}} \left(u_{i\rho_q} y_{i\rho_q} + u_{\rho_q i} y_{\rho_q i} \right) \right\} \tag{3}$$

subject to

$$x_{ij} \geq x^k_{ij} + x^k_{ji} \quad \forall i, j \in \rho_r, (i, j) \in E, \forall k \in Z_d \tag{4a}$$

$$y_{i\rho_q} \geq y^k_{i\rho_q} \quad \forall i \in \rho_r, \rho_q \in \Theta \setminus \{\rho_r\}, \quad \forall k \in Z_d \tag{4b}$$

$$y_{\rho_q i} \geq y^k_{\rho_q i} \quad \forall i \in \rho_r, \rho_q \in \Theta \setminus \{\rho_r\}, \quad \forall k \in Z_d \tag{4c}$$

$$\left(\sum_{i,j \in \rho_r} x^k_{ij} + \sum_{\rho_q \in \Theta \setminus \{\rho_r\}} y^k_{i\rho_q} \right) - \left(\sum_{h,i \in \rho_r} x^k_{hi} + \sum_{\rho_q \in \Theta \setminus \{\rho_r\}} y^k_{\rho_q i} \right) = \begin{cases} 1, & \text{if} \quad i = s \\ -1, & \text{if} \quad i = k \\ 0, & \text{otherwise} \end{cases}$$
$$\forall i \in \rho_r, \forall k \in Z_d \tag{4d}$$

$$\left(\sum_{i \in \rho_r} y^k_{\rho_q i} + y^k_{\rho_q d_r} \right) - \left(\sum_{i \in \rho_r} y^k_{i\rho_q} + y^k_{d_r \rho_q} \right) = \begin{cases} 1, & \text{if} \quad s \in \rho_q, k \notin \rho_q \\ -1, & \text{if} \quad k \in \rho_q, s \notin \rho_q \\ 0, & \text{otherwise} \end{cases}$$
$$\forall k \in Z_d; \rho_q \in \Theta \setminus \{\rho_r\} \tag{4e}$$

$$0 \leq x_{ij} \leq 1 \quad \forall i, j \in \rho_r, \ (i, j) \in E \tag{4f}$$

$$0 \leq y_{i\rho_q} \leq 1 \quad \forall i \in \rho_r, \rho_q \in \Theta \setminus \{\rho_r\} \tag{4g}$$

$$0 \leq y_{\rho_q i} \leq 1 \quad \forall i \in \rho_r, \rho_q \in \Theta \setminus \{\rho_r\} \tag{4h}$$

$$x_{ij} \leq \lfloor b_{ij}/f \rfloor \quad \forall i, j \in \rho_r, \ (i, j) \in E \tag{4i}$$

All variables are non-negative integers. (4j)

The constraints (4a) to (4h) ensure locally that a possible route is a tree which connects the nodes in Z. Constraints (4a) to (4c) are extention of (2a). Constraints (4d) and (4e) are extension of (2b). (4f) correspond to (2c) while (4g) are new constraints specific to aggregate links. (4i) guarantee that only those links with enough capacities can be used to accomodate the given flow. P_1 is a mixed integer programming problem in terms of multicommodity flows, where each flow corresponds to an embeded path in the tree from the source to one of the destinations.

Given clusters r,q, and aggregate flows $\boldsymbol{y_{rq}}$ between these clusters, the objective function of P_2 is to minimize the sums of the flows costs on the links between cluster r and cluster q and the flow costs on articifial links between cluster r or q and the dummy node associated with it. The unit flow costs on the articifial links are assigned large positive integer values, implying that the given flows from one node to the corresponding aggregate node are preferred to passing through the links existing between them, rather than traversing any other intermediate nodes in these clusters.

$P_2((r, q), \boldsymbol{y_{rq}})$:

$$\text{Min} \left\{ \sum_{\substack{i \in \rho_{r \to q} \\ j \in \rho_{q \to r} \\ (i,j) \in E}} c_{ij} x_{ij} + M \times \left(\sum_{i \in \rho_{r \to q}} \left(x_{id^1_{rq}} + x_{d^1_{rq} i} \right) + \sum_{j \in \rho_{q \to r}} \left(x_{jd^2_{rq}} + x_{d^2_{rq} j} \right) \right) \right\} \tag{5}$$

subject to

$$\sum_{\substack{j\in\rho_{q\to r}\\(i,j)\in E}} x_{ij} + x_{id^1_{rq}} - x_{d^1_{rq}i} = y_{i\rho_q} \quad \forall i \in \rho_{r\to q} \quad (6a)$$

$$\sum_{\substack{i\in\rho_{r\to q}\\(i,j)\in E}} x_{ij} + x_{d^2_{rq}j} - x_{jd^2_{rq}} = y_{\rho_r j} \quad \forall j \in \rho_{q\to r} \quad (6b)$$

$$\sum_{i\in\rho_{r\to q}} (x_{id^1_{rq}} - x_{d^1_{rq}i}) = 0 \quad (6c)$$

$$\sum_{j\in\rho_{q\to r}} (x_{jd^2_{rq}} - x_{d^2_{rq}j}) = 0 \quad (6d)$$

$$0 \leqq x_{ij} \leqq 1 \quad \forall i \in \rho_{r\to q}, \forall j \in \rho_{q\to r}, (i,j) \in E \quad (6e)$$

$$0 \leqq x_{id^1_{rq}} \leqq 1 \quad \forall i \in \rho_{r\to q} \quad (6f)$$

$$0 \leqq x_{d^1_{rq}i} \leqq 1 \quad \forall i \in \rho_{r\to q} \quad (6g)$$

$$0 \leqq x_{jd^2_{rq}} \leqq 1 \quad \forall j \in \rho_{q\to r} \quad (6h)$$

$$0 \leqq x_{d^2_{rq}j} \leqq 1 \quad \forall j \in \rho_{q\to r} \quad (6i)$$

$$x_{ij} \leqq \lfloor b_{ij}/f \rfloor \quad \forall i \in \rho_{r\to q}, \forall j \in \rho_{q\to r}, (i,j) \in E \quad (6j)$$

All variables are non-negative integers. (6k)

Equations (6a) and (6b) ensure that the given aggregate flows are enforced on the relevant links. Equations (6c) and (6d) ensure that the sum of flows entering and going out from each dummy node is zero, implying no flows arrive at the nodes which are not destinations of the given aggregate flows. The upper bounds for the flows on artificial links are also one, as specified in constraints (6f) to (6i).

The original CBMR problem can then be reformulated based upon these subproblems. The objective function of the reformulated problem can be written as

$$\min_{(4a)\text{-}(4j)} \left\{ \sum_r \sum_{\substack{i,j\in\rho_r\\(i,j)\in E}} c_{ij}x_{ij} + \sum_r \sum_{q(q\neq r)} P_2^{opt}((r,q), y_{rq}) \right\}, \quad (7)$$

where $P_2^{opt}((r,q),y_{rq})$ denotes the optimum solution of $P_2((r,q),y_{rq})$.

The problem is an integer linear programming (ILP) problem. The solving method of linear programming (LP) problems is often used to help solve ILP problems due to its good efficiency (it only needs polynomial time[16]). Although, in general cases, the optimum solution of the linear programming (continuous) relaxation of an ILP problem must not the optimum solution of the ILP problem, the former can be used as the bound of the latter.

Now we relax this problem, by introducing linear programming relaxations of P_1 and P_2, i.e., LP_1 and LP_2, which are obtained by removing their integer constraints (4j) and (6k) respectively. The relaxation will not change the integral property of solutions according to the following principle shown in [15].

Integral Property: *For minimum cost flow problems having integer bounds and demands, all basic variables (including the optimum one) also have integer values.*

Thus, the relaxation version of the objective function, which is equivalent to the objective function of (7), can then be written as

$$\min_{(4a)\text{-}(4i)} \left\{ \sum_r \sum_{\substack{i,j\in\rho_r\\(i,j)\in E}} c_{ij}x_{ij} + \sum_r \sum_{q(q\neq r)} LP_2^{opt}((r,q), y_{rq}) \right\},$$

where $LP_2^{opt}((r,q),y_{rq})$ denotes the optimum solution of $LP_2((r,q),y_{rq})$.

Using the dual of LP_2, i.e., DLP_2, which is shown in Appendix, we can rewrite this objective function as (because $LP_2^{opt}((r,q),y_{rq}) = DLP_2^{opt}((r,q),y_{rq})$)

$$\min_{(4a)\text{-}(4i)} \left\{ \sum_r \sum_{\substack{i,j\in\rho_r\\(i,j)\in E}} c_{ij}x_{ij} + \sum_r \sum_{q(q\neq r)} DLP_2^{opt}((r,q), y_{rq}) \right\},$$

where $DLP_2^{opt}((r,q),y_{rq})$ denotes the optimum solution of $DLP_2((r,q),y_{rq})$.

Further, using variable vectors, the above objective function can be written as

$$\min_{(4a)\text{-}(4i)} (\boldsymbol{c}\cdot\boldsymbol{x} + \max_{(12a)\text{-}(12h)} (\boldsymbol{v}\cdot\boldsymbol{y} + \boldsymbol{w}\cdot\boldsymbol{b})),$$

that is,

$$\min_{(4a)\text{-}(4i)} (\boldsymbol{c}\cdot\boldsymbol{x}) + \min_{(4a)\text{-}(4i)} \max_{(12a)\text{-}(12h)} (\boldsymbol{v}\cdot\boldsymbol{y} + \boldsymbol{w}\cdot\boldsymbol{b}).$$

This is equivalent to (TOS is the abbreviation of The Optimum Solution):

$$\max_{(12a)\text{-}(12h)} (\min_{(4a)\text{-}(4i)} (\boldsymbol{c}\cdot\boldsymbol{x} + \boldsymbol{v}\cdot\boldsymbol{y}) + \boldsymbol{w}\cdot\boldsymbol{b}) = \text{TOS}$$

and

$$\min_{(4a)\text{-}(4i)} (\max_{(12a)\text{-}(12h)} (\boldsymbol{v}\cdot\boldsymbol{y} + \boldsymbol{w}\cdot\boldsymbol{b}) + \boldsymbol{c}\cdot\boldsymbol{x}) = \text{TOS}$$

That is,

$$\max_{(12a)\text{-}(12h)} \left(\sum_r LP_1^{opt}(r, \boldsymbol{v}_r) + \boldsymbol{w}\cdot\boldsymbol{b} \right) = \text{TOS} \quad (8)$$

and

$$\min_{(4a)\text{-}(4i)} \left(\sum_{\substack{r,q\\r\neq q}} DLP_2^{opt}((r,q), y_{rq}) + \boldsymbol{c}\cdot\boldsymbol{x} \right) = \text{TOS} \quad (9)$$

According to (8) and (9), it is possible to find the lower and upper bounds for the optimum solution. By diminishing the gap of these bounds, the optimum solution can be obtained asymptotically.

3.3 Routing Algorithm

In this section, we present an algorithm for CBMR based on the problem reformulation described previously. According to (8) and (9), our task is to find the lower and upper bounds for the optimum solution. By reducing the gap of the two bounds, an asymptotical optimum solution can be obtained. In particular, if the gap becomes zero, it means the optimum solution has been found. The algorithm consists of three phases: the first phase is to determine the routing nodes to be involved in making routing decisions, and to initialize some variables, the second phase is to use an optimization loop to get the optimum solution, the third phase is to yield the final solution. As an input parameter, ε is given to stop running of the algorithm (i.e., to exit the optimization loop).

The algorithm is realized in a distributed cooperative manner. A center node needs to be selected to collect local computation results and do calculations to obtain the lower and upper bounds. During the running of the algorithm, the costs of aggregate links are updated to reflect any decision made by local routing nodes. The updated costs are notified to the routing nodes of the corresponding clusters. For instance, $v_{\rho_r j}$ ($v_{i\rho_q}$) is updated by the routing node of cluster r (q) and its new value is sent to the routing node of the cluster to which j (i) belongs. The new values are then used in making routing decisions in the next iteration.

In the algorithm, we discriminate clusters using the following terms.

Source cluster: A cluster where the source node of the multicast group is contained.

Destination cluster: A cluster which contains only destination nodes of the multicast group.

Given $\Theta = \{\rho_1, \rho_2, \cdots, \rho_m\}$ (the set of clusters which contain multicast members), we suppose ρ_1 represent the source cluster and the others represent destination clusters.

Algorithm:
Phase 1 - *Initialization*

(1) Determine the center node and the routing nodes (each routing node is selected from a cluster in Θ. We can determine the routing node in the source cluster as the center node). Confirm the neighborhood among these nodes and establish necessary communication channels between them. Set aggregate link costs $u_{i\rho_q} = 0$, $u_{\rho_q i} = \infty$ ($\forall i \in \rho_r, \forall \rho_r \in \Theta$, and $\forall \rho_q \in \Theta \setminus \{\rho_r\}$).

(2) In the source cluster, solve $LP_1(1, \boldsymbol{u}_1)$. This yields a tree among the multicast member nodes in the source cluster and its adjacent aggregate nodes. The aggregate link costs $u_{\rho_1 j}(\forall j \in \rho_q, \rho_q \in \Theta \setminus \{\rho_1\})$ are obtained by calculating the minimum-cost path between j and the resultant tree. Send $u_{\rho_1 j}$ to the routing node of the cluster to which j belongs.

(3) In destination cluster q ($q = 2, 3, \cdots, m$), update $u_{\rho_r j}(\rho_r \in \Theta \setminus \{\rho_q\}, j \in \rho_q)$ if new value was received from the routing node of cluster r. If there is a link between node j of cluster q and node k of cluster p ($\rho_p \in \Theta \setminus \{\rho_r, \rho_q\}$), let $u_{\rho_q k} = u_{\rho_r j} + c_{jk}$ and send it to the routing node of cluster p. Solve $LP_1(q, \boldsymbol{u}_q)$. This yields $(\boldsymbol{x}_q^{\mathrm{o}}, \boldsymbol{y}_q^{\mathrm{o}})$. Send the elements of $\boldsymbol{y}_q^{\mathrm{o}}$ to the routing nodes of related clusters. Let $\boldsymbol{x}_q(0) = \boldsymbol{x}_q^{\mathrm{o}}, \boldsymbol{y}_q(0) = \boldsymbol{y}_q^{\mathrm{o}}$.

(4) Solve $DLP_2((r, q), \boldsymbol{y}_{rq})$ in the routing node of cluster q if $\boldsymbol{y}_{rq} \neq \mathbf{o}$. This yields dual extreme solutions $(\boldsymbol{v}_{rq}^1, \boldsymbol{w}_{rq}^1)$. Send the non-zero elements of $\boldsymbol{v}_{rq}^1$ to the routing node of cluster r, $\boldsymbol{w}_{rq}^1$ to the center node. Set $\boldsymbol{v}_r(0) = \mathbf{o}, \boldsymbol{w}_r(0) = \mathbf{o}$. In the center node, set iteration number $t = 1$, $LB(0) = -\infty$, $UB(0) = \infty$.

Phase 2 - *Optimization Loop*

(5) In the source cluster, solve $LP_1(1, \boldsymbol{v}_1(t-1))$. This yields solution $(\boldsymbol{x}_1^t, \boldsymbol{y}_1^t)$. The costs $u_{\rho_1 j}(\forall j \in \rho_q, \rho_q \in \Theta \setminus \{\rho_1\})$ are updated by calculating the minimum-cost path between j and the resultant tree. Send $u_{\rho_1 j}$ to the routing node of the cluster to which j belongs.

(6) In destination cluster q ($q = 2, 3, \cdots, m$), if non-zero elements exist in $\boldsymbol{v}_q^t$ or $\boldsymbol{w}_q^t$, update vectors $\boldsymbol{v}_q(t)$ and $\boldsymbol{w}_q(t)$ as follows (only for those elements while the corresponding $\boldsymbol{v}_q^t$'s or $\boldsymbol{w}_q^t$'s elements are not zeros).

$$\boldsymbol{v}_q(t) = ((t-1)/t)\boldsymbol{v}_q(t-1) + (1/t)\boldsymbol{v}_q^t \ ,$$

$$\boldsymbol{w}_q(t) = ((t-1)/t)\boldsymbol{w}_q(t-1) + (1/t)\boldsymbol{w}_q^t \ .$$

Solve $LP_1(r, \boldsymbol{v}_q(t))$. This yields $(\boldsymbol{x}_q^t, \boldsymbol{y}_q^t)$. Send $\boldsymbol{y}_q^t$ to the routing nodes of related clusters. The costs $u_{\rho_q j}(\forall j \in \rho_r, \rho_r \in \Theta \setminus \{\rho_q\}, y_{\rho_q j} = 1)$ are obtained by calculating the minimum-cost path between j and the subtree consisting of non-zero elements of $\boldsymbol{x}_q^t$. Send $\boldsymbol{c} \cdot \boldsymbol{x}_q^t$ and $\sum_{y_{\rho_q j}=1} u_{\rho_q j} y_{\rho_q j}$ to the center node. In the above, $\boldsymbol{v}_q$ and $\boldsymbol{w}_q$ are set to be the sums of their previous values multiplied by the numbers of their occurrence (utilizing the information obtained so far). Through this approach, the values of the vector elements get closer to the values (*stable points*) which occurred most frequently.

(7) In cluster q, solve $DLP_2((r,q), y^t_{rq})$ to obtain dual vectors $(v^{t+1}_{rq}, w^{t+1}_{rq})$. Send the non-zero elements of v^{t+1}_{rq} to the routing node of cluster r, w^{t+1}_{rq} to the center node.

(8) In the center node, update the lower bound for the optimum solution as

$$LB(t) = \mathrm{Max}\left\{\left(\sum_r LP^{opt}_1(r, v_r) + w \cdot b\right), LB(t-1)\right\}$$

while the upper bound for the optimum solution is updated as

$$UB(t) = \mathrm{Min}\left\{\left(\sum_{r,q} DLP^{opt}_2((r,q), y_{rq}) + c \cdot x\right), UB(t-1)\right\},$$

where $c \cdot x = \sum_q (c \cdot x^t_q + u_{\rho_q j} y_{\rho_q j})$.

Set $x_r(t) = x^t_r$, $y_{rq}(t) = y^t_{rq}$ $(\forall r, q \in \Theta, r \neq q)$ and $t = t + 1$ if $UB(t) < UB(t-1)$.

If $[(UB(t) - LB(t))/UB(t)] > \varepsilon$ and $LB(t) \neq LB(t-1)$, go to step (5). Otherwise, send a stop message to other routing nodes.

Phase 3 - *Ending*

(9) Constitute subgraphs with links of non-zero elements of $x_r(t)$ and non-zero disaggregated elements of $y_{rq}(t)$. This produces the final solution of the problem.

In the algorithm, the dual variable vector v_q represents the costs between multicast members in different clusters. The minimum values of v_q's elements equal to the costs of inter-cluster links. For instance, the minimum value of $v_{i\rho_r}$ $(i \in \rho_q, j \in \rho_r, q \neq r)$ equals to c_{ij} while nodes i, j are multicast members which can be connected with a direct link between them. Meanwhile, the elements of v_q have upper bounds since the cost values of paths between one node to any multicast members in neighboring clusters are bounded.

We specify more details relating with the algorithm. First, we explain the lower and upper bounds set in step (8). From the formulation of DLP_2, we can see that the dual variables of $w \cdot b$ must be non-positive numbers. In other words, the item $w \cdot b$ in $LB(t)$ setting will not larger than zero. We only need to be the aware of the first item, $\sum_r LP^{opt}_1(r, v_r)$. The lower bound expressed like this also reflects the fact that the individual local solutions do not produce a global tree. On the other hand, from the setting of $UB(t)$, the upper bound here implies a global tree because it connects all the multicast members together. It is a feasible solution of the original problem. Thus, the final solution can be obtained from the upper bound.

Next, we describe the complexities of the algorithm. Before that, we have a proposition below.

Proposition 1: The routing algorithm stops within limited iterations if the link costs are rational numbers.

Proof: As stated previously, the values of v_q's elements are bounded between certain numbers that are related with the spectrum of link costs. We assume that the objective function takes values having the same number of decimals with the link costs. We change v_q according to $v_q(t) = ((t-1)/t)v_q(t-1) + (1/t)v^t_q$ iteratively. Moreover, the total number of elements is finite. Thus, the vector elements can reach their stable points $(LB(t) = LB(t-1))$ within limited iterations. □

Let n denote the cluster size (number of nodes in a cluster), m the number of clusters in the object graph, μ the maximum connectivity (maximum number of links) of a node, ν the maximum connectivity (maximum number of links) between two clusters, p the number of multicast members. According to [16], the computational complexities of LP_1 and DLP_2 are $O(n^4 p^4 (\mu + m)^4 \log np(\mu+m))$ and $O(\nu^4 \log \nu)$ respectively. Solving minimum-cost paths requires $O(n^2)$. They have smaller orders than LP_1. Because the algorithm stops within limited iterations, the computational complexity of the algorithm can be counted by the complexity of one iteration, i.e., $O(n^4 p^4 (\mu + m)^4 \log np(\mu + m))$.

As a measure indicating the amount of messages transferred between routing nodes, the communication complexity is $O(mn)$.

Proposition 2: The routing algorithm is convergent.

Proof: It is ensured by setting $UB(t)$ as in step (8) that the solution obtained will descend. As Proposition 1 says, the algorithm stops within limited iterations. When $LB(t) = LB(t-1)$, it is not possible to increase LB any more. In other words, LP_1 reached stable points. We use step (8) to enforce the algorithm to select the best one from possible stable points' set. The algorithm yields optimum or near-optimum solutions. □

4. Numerical Results

4.1 Network Model

Networks with mesh topologies are chosen to be studied in the simulation. We adopt a method similar to that used in [3] to generate the topology. For N nodes that are randomly located on cross points of a rectangular grid, the Euclidean distances are used to represent the distances between node pairs. A link is introduced between node pairs depending on the link probability

of

$$P\{(i,j)\} = \frac{\gamma\delta}{N}\beta \exp\left(\frac{-d_{ij}}{\alpha D}\right), \tag{10}$$

where d_{ij} is the distance between nodes i and j. D is the maximum distance between two nodes in the graph. β is the parameter determining the density of the graph. α is used to adjust the connectivity of a node with other nodes; while α gets small, short links get dense compared with long links. δ is the mean node connectivity. γ is related with α and β. For instance, when $\alpha = 0.25$, $\beta = 0.47$, γ is approximately 12.

The available capacity of each link is assigned according to a uniform random distribution in the range of (0, B]. Link cost is set to the inverse number (letting the value have three decimals) of available capacity of that link.

4.2 Simulation Results

We consider networks constituted of comparatively large number of nodes. In the experiment, we set $\delta = 8, B = 10$. Suppose the capacity required by the multicast connection is 1. The multicast connection needs to be established among the nodes of a multicast group, which are selected randomly out of the network. Clusters are generated following the principles described in Sect. 2 when the value of cluster size is given. After multicast member nodes are given, the object graph can then be determined.

The linear programming subproblems are solved by an LP solver called *lpsolve*. The simulations are carried out on Sun SS-20. Four main issues are investigated through the simulation. All simulations are conducted with multiple cases and the resultant data are taken from their averages. ε is set to be 0.05. We should set it properly (not extremely small) in order to tolerate the cumulative d eviation caused by the operation (round-off) on variables of $\boldsymbol{v}$.

(a) Convergence

As an example, we let $N = 120$, and let the cluster size vary in the way that just divides the network uniformly. Figure 4 shows the convergence of the solutions while the algorithm is executed.

The solution obtained at each iteration is compared with the optimum solution, which is obtained by using an enumeration method. The cost ratio is defined as the cost of the resultant multicast tree to that of the optimum, indicating the optimality of a solution. The data are obtained when the multicast group size (number of nodes in a multicast) is set to be 6.

From Fig. 4 (The numbers in brackets after the cluster sizes indicate the mean connectivity between clusters), we can see that the resultant solutions are getting near to the optimum while the number of iterations increases. The fact of asymptotical feature can be observed from the trend of lines in the figure. We should

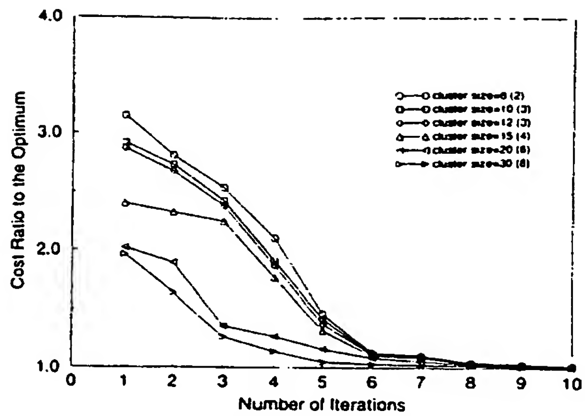

Fig. 4 Convergence on iterations ($N = 120$, $\alpha = 0.25$, $\beta = 0.47$, $\gamma = 12$, $\delta = 8$).

note that, however, the algorithm will not definitely yield the optimum solution because in the algorithm round-off operations are needed for the variables of $\boldsymbol{v}$.

(b) The relation between solution optimality and algorithm's running time

Sometimes, the number of iterations is not obvious for us to determine the efficiency of the algorithm. Moreover, under the circumstance of limited running time, the corresponding running time is the most appropriate parameter determining the efficiency.

Figure 5 shows the relation between solution optimality and algorithm's running time. The parameters used are the same with the above. In each iteration, the time needed to run the algorithm is measured. The times running concurrently executable steps are not added together and only the maximum one is taken account of.

The efficiency of the algorithm depends on appropriate cluster size. According to the structure of the algorithm, it can be stopped within given time. This will result in approximately optimum solutions. In each iteration, we measure the time needed to run the algorithm, including the communication time needed to exchange messages between routing nodes. We set the time for transmitting one message between two routing nodes is 0.1 second (same afterward) herein.

(c) Various network sizes

The scalability of the algorithm is investigated by changing the total number of nodes in the network (network size). Figure 6 shows the running time needed for various number of nodes. Four kinds of network sizes 60, 100, 300, 1000 are tested. The experiment is conducted by setting the multicast group size to be 6, 10, 20 and 30 respectively.

In the figure, when the network size becomes large, only moderate increase of running time was observed. Also, there was no fast increase in the running time while the multicast group size gets large. Along with the complexity analysis in Sect. 3.3, we can anticipate

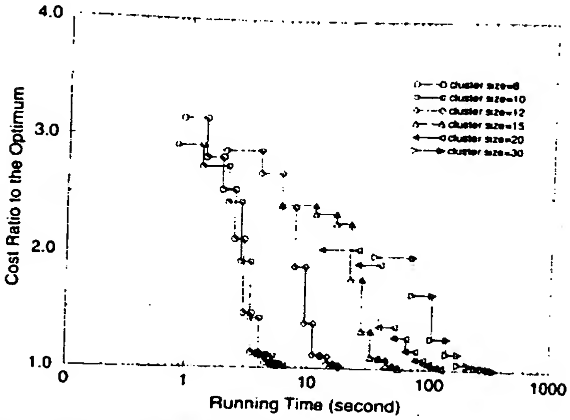

Fig. 5 Relation between solution optimality and running time ($N = 120$, $\alpha = 0.25$, $\beta = 0.47$, $\gamma = 12$, $\delta = 8$).

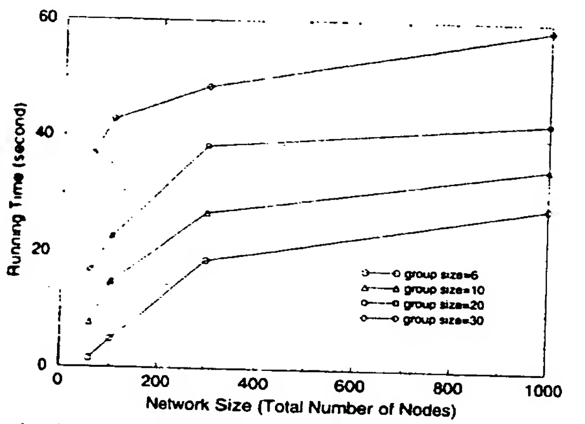

Fig. 6 Running time for various network sizes (cluster size = 10).

that this approach is efficient for large networks.

(d) Comparison with other method

Since our approach finds optimum multicast trees asymptotically, it yields solutions efficiently. We compare it with an algorithm called Steiner Tree Enumeration Algorithm (STEA) [1], which is considered to be efficient for the routing of multipoint connection in a small local network. In our experiment, the multicast trees among clusters are previously searched using a tree enumeration algorithm, and then STEA is used to find local solutions within clusters.

Figure 7 shows the comparison of time efficiency by using the two algorithms under the same conditions. The network size is set to be 200 and multicast group size is 10. Cluster size is set to be 5, 10, 20, 30 and 50 respectively. Since in our approach the lower and upper bounds are obtained based on aggregate and disaggregate technique (ADT), we can determine that near-optimum solution is obtained once the gap of the bounds gets small enough. However, by STEA we cannot do like this. We have to search over all the cases

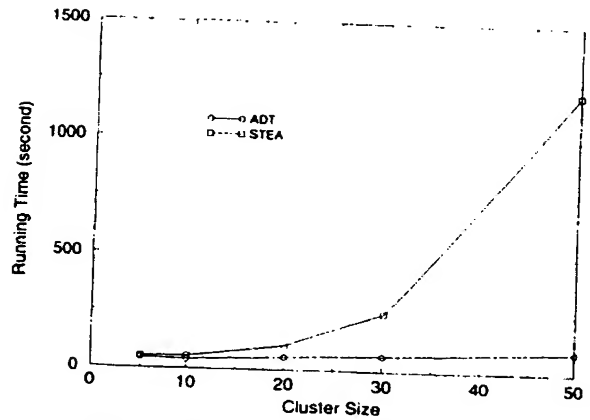

Fig. 7 Comparison with STEA ($N = 200$).

first, then make decisions. The difference appears apparently, especially when the cluster size gets large. In addition, since we set ε to be small(0.05), our algorithm achieved results with the cost ratio to optimum below 1.03. This turns out that the algorithm has good optimization performance as well.

4.3 Summary

The simulation results show that solution convergence can be achieved by the presented algorithm. It yields near-optimum solutions within moderate running time. Meanwhile, cluster size will influence the efficiency of the algorithm. It suggests that while using the algorithm the cluster size should be appropriately selected so as to obtain the solution in an efficient way. Scalability can be realized while network size becomes large and the multicast group size is relatively large. Finally, the presented approach shows higher time efficiency than STEA in finding optimum solutions.

With different network topologies, which determined by the values of α, β and δ, the efficiency in finding solutions will change somewhat. We omit the details on this aspect, and leave this kind of investigation as a work in studying practical systems.

5. Conclusion

In this paper, we proposed the CBMR approach for hierarchical multicast routing. In our scheme, as a condition, only incomplete information is available to routing nodes. The objective is to find the asymptotical optimum solution for routing of multipoint connection. We have shown by computer simulation that the routing approach yields reasonable efficiency in terms of running time and solution optimality. The routing approach can compromise the application requirement based on the trade-offs existing between the running time and the solution optimality. Such requirement can be seen in most practical systems.

The presented approach has two main characteristics, 1) it yields asymptotical optimum solutions for the routing of multipoint connection, 2) the routing decisions can be made in the environment where only partial information is available to routing nodes. These characteristics are beneficial to multicast routing in large-scale networks. Some extension should be made if the approach is required to deal with multicast routing problems where additional requests such as when delay constraints are added. On the other hand, while the requirement of time efficiency is more important than that of the solution optimality, some other techniques such as heuristics would be required in finding approximate solutions.

References

[1] P. Winter, "Steiner problem in networks: A survey," NETWORKS, vol.17, pp.129–167, 1987.

[2] K. Bharath-Kumar and J.M. Jaffe, "Routing to multiple destinations in computer networks," IEEE Trans. Commun., vol.COM-31, no.3, pp.343–351, March 1983.

[3] B.M. Waxman, "Routing of multipoint connections," IEEE J. Select. Areas Commun., vol.6, no.9, pp.1617–1622, Dec. 1988.

[4] C.A. Noronha, Jr. and F.A. Tobagi, "Optimum routing of multicast streams," Proc. IEEE INFOCOM'94, pp.865–873, 1994.

[5] V. Kompella, J. Pasquale, and G. Polyzos, "Multicast routing for multimedia communication," IEEE/ACM Trans. Networking, vol.1, no.3, pp.286–292, June 1993.

[6] P.C. Huang and Y. Tanaka, "Multicast routing based on predicted traffic statistics," IEICE Trans. Commun., vol.E77-B, no.10, pp.1188–1193, Oct. 1994.

[7] V. Priwan, H. Aida, and T. Saito, "The multicast tree based routing for the complete broadcast multipoint-to-multipoint communications," IEICE Trans. Commun., vol.E78-B, no.5, pp.720–728, May 1995.

[8] S. Ramanathan, "Multicast tree generation in networks with asymmetric links," IEEE/ACM Trans. Networking, vol.4, no.4, pp.558–568, Aug. 1996.

[9] T. Ballardie, P. Francis, and J. Crowcroft, "Core Based Trees (CBT): An architecture for scalable inter-domain multicast routing," Proc. SIGCOMM'93, pp.85–95, Sept. 1993.

[10] F. Kamoun and L. Kleinrock, "Stochastic performance evaluation of hierarchical routing for large networks," Computer Networks, vol.3, pp.337–353, 1979.

[11] C. Galand and P. Scotton, "Automatic network clustering for fast path selection," High Performance Networking, IFIP Transactions V(C-26), pp.123–133, 1994.

[12] E. Balas, "Solution of large scale transportation problems through aggregation," Operations Res., vol.13, pp.82–93, 1965.

[13] L.S. Lasdon, "Optimization Theory for Large Systems," Macmillan, 1970.

[14] A.V. Iyer, J.J. Jarvis, and H.D. Ratliff, "Hierarchical solution of network flow problems," NETWORKS, vol.20, pp.731–752, 1990.

[15] F.S. Hillier and G.J. Lieberman, "Introduction to Operations Research," p.374, McGraw-Hill, 1995.

[16] M.S. Bazaraa, J.J. Jarvis, and H.D. Sherali, "Linear Programming and Network Flows," ch.8, John Wiley & Sons, 1990.

Appendix

Given an LP problem in the form of

$$\text{Min } c^T x$$

subject to

$$Ax = b$$

$$x \geqq o \, .$$

Its dual takes the form of

$$\text{Max } \psi^T b$$

subject to

$$A^T \psi \leqq c \, .$$

According to this, by adding slack variables into inequalities (6e) to (6j), we can dualize LP_2 as follows.

$$DLP_2((r,q), y_{rq}):$$

$$\text{Max} \left\{ \sum_{i \in \rho_{r \to q}} v_{i\rho_q} y_{i\rho_q} + \sum_{j \in \rho_{q \to r}} v_{\rho_r j} y_{\rho_r j} + \sum_{\substack{i \in \rho_{r \to q} \\ j \in \rho_{q \to r}}} w_{ij} \lfloor b_{ij}/f \rfloor + \sum_{\substack{i \in \rho_{r \to q} \\ j \in \rho_{q \to r}}} \left(w_{id^1_{rq}} + w_{d^2_{rq} j} \right) \right\} \tag{A-1}$$

subject to

$$v_{i\rho_q} + v_{\rho_r j} + w_{ij} \leqq c_{ij}$$
$$\forall i \in \rho_{r \to q}, \forall j \in \rho_{q \to r}, (i,j) \in E \tag{A-2a}$$
$$v_{i\rho_q} + v_{d^1_{rq}} + w_{id^1_{rq}} \leqq M \qquad \forall i \in \rho_{r \to q} \tag{A-2b}$$
$$-v_{i\rho_q} - v_{d^1_{rq}} + w_{id^1_{rq}} \leqq M \qquad \forall i \in \rho_{r \to q} \tag{A-2c}$$
$$-v_{\rho_r j} + v_{d^2_{rq}} + w_{d^2_{rq} j} \leqq M \qquad \forall j \in \rho_{q \to r} \tag{A-2d}$$
$$v_{\rho_r j} - v_{d^2_{rq}} + w_{d^2_{rq} j} \leqq M \qquad \forall j \in \rho_{q \to r} \tag{A-2e}$$
$$w_{ij} \leqq 0 \qquad \forall i \in \rho_{r \to q}, \forall j \in \rho_{q \to r}, (i,j) \in E \tag{A-2f}$$
$$w_{id^1_{rq}} \leqq 0 \qquad \forall i \in \rho_{r \to q} \tag{A-2g}$$
$$w_{d^2_{rq} j} \leqq 0 \qquad \forall j \in \rho_{q \to r} \, . \tag{A-2h}$$

Yibo Zhang received the B.E. degree in computer science from Tsinghua University, Beijing, China, in 1987, the M.E. and Ph.D. degrees in electronic engineering from University of Tokyo, in 1994 and 1998 respectively. His research interests include routing, flow control, and distributed algorithms. He joined the R&D Laboratory, Uniden Corporation, Tokyo, Japan, in April 1998.

Weiping Zhao received the Bachelar's degree in telecommunication systems from Beijing University of Posts and Telecommunications, China, in 1982, the Master degree in telecommunication systems from Essex University, U.K., in 1987 and the Ph.D. degree in electronic engineering from University of Tokyo in 1993. He joined the National Center for Science Information Systems (NACSIS) in 1993. His research interests include modeling and analysis of communication systems, congestion control in broadband networks.

Shunji Abe graduated from Toyohashi University of Technology, Department of Information Engineering, in 1980 and received the M.S. degree in 1982. In that year, he joined Fujitsu Laboratories LTD. Since then he has engaged in research and development of B-ISDN switching systems and traffic control systems. Since June 1995 he has been with the National Center for Science Information Systems (NACSIS). He received the D.Eng. degree from University of Tokyo in 1996. He is a member of IEEE.

Shoichiro Asano had graduated at Electronic Engineering, Faculty of Engineering, University of Tokyo, in 1970. He received ME and DE, both from University of Tokyo, in 1972 and 1975 respectively. Dr. Asano is a Professor at the National Center for Science Information Systems (NACSIS) and a Professor at University of Tokyo. His researches are mainly focused on digital integrated network architecture and development of Science Information Network in Japan. He is a vice chairman of Information, Computer, Communication Policy (ICCP) at OECD.